

## Graph colouring MAC protocol for underwater sensor networks

Alfouzan, Faisal Abdulaziz F; Shahrabi, Alireza; Ghoreyshi, Seyed Mohammad; Boutaleb, Tuleen

*Published in:*  
IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)

*DOI:*  
[10.1109/AINA.2018.00030](https://doi.org/10.1109/AINA.2018.00030)

*Publication date:*  
2018

*Document Version*  
Author accepted manuscript

[Link to publication in ResearchOnline](#)

*Citation for published version (Harvard):*  
Alfouzan, FAF, Shahrabi, A, Ghoreyshi, SM & Boutaleb, T 2018, Graph colouring MAC protocol for underwater sensor networks. in *IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*. IEEE, pp. 120-127. <https://doi.org/10.1109/AINA.2018.00030>

### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

### Take down policy

If you believe that this document breaches copyright please view our takedown policy at <https://edshare.gcu.ac.uk/id/eprint/5179> for details of how to contact us.

# Graph Colouring MAC Protocol for Underwater Sensor Networks

Faisal Alfouzan, Alireza Shahrabi, Seyed Mohammad Ghoreyshi, and Tuleen Boutaleb

School of Engineering and Built Environment

Glasgow Caledonian University, Glasgow, United Kingdom

{Faisal.Alfouzan, A.Shahrabi, Seyed.MohammadGhoreyshi, T.boutaleb}@gcu.ac.uk

**Abstract**—Employing contention-based MAC protocols in underwater sensor networks are typically costly. This is mainly due to the unique characteristics of its acoustic channels such as long propagation delay, high bit error rate, and limited bandwidth. As a consequence, handshake-based and random access-based MAC protocols do not perform as well as expected. The collision-free approach is therefore considered to achieve a better performance by efficiently addressing spatial-temporal uncertainty, hidden/exposed terminal problems, and near-far effect at MAC layer, thus collisions and retransmissions are properly avoided in order to reduce the energy cost and also to improve the throughput and fairness across the network. In this paper, we propose a novel energy-conserving and collision-free graph colouring MAC protocol, called GC-MAC, for UWSNs. It employs a TDMA-like principle by assigning separate time slots to every individual colour in the network. Nodes with the same colours can thus transmit concurrently without any collision. GC-MAC also does not require CDMA or power adjustment for collision resolution. Our extensive simulation study reveals that our proposed protocol can efficiently handle the traffic contention to achieve significant improvement in terms of throughput, energy consumption, and fairness index under varying offered loads.

**Index Terms**—Underwater sensor networks (UWSNs), Media access control (MAC), Graph colouring, Distributed clustering approach.

## I. INTRODUCTION

Due to the wide range of applications, such as environmental monitoring, oceanographic data collection, early warning systems, tactical surveillance, assisted navigation, and resource discovery, underwater sensor networks (UWSNs) have attracted significant attention over the last decade [1–3]. Due to some inherent characteristics of underwater acoustic channels, such as long propagation delays, limited bandwidth, and low data transmission rates [4], the design of MAC (Medium Access Control) protocols in UWSNs faces many challenges.

The MAC protocol can generally be divided into two categories: contention-free and contention-based protocols [5]. In the contention-free protocol, communication channels are separated into time, frequency or code domains, such as Time Division Multiple Access (TDMA), Frequency Division Multiple Access (FDMA), and Code Division Multiple Access (CDMA) [6]. Contention-based protocols include handshaking-based and random access-based MAC protocols. In the handshaking-based protocol, the sender and receiver capture the medium through control packet exchange before data transmission, while in a random access-based protocol, the sender sends packets without coordination. When a data packet arrives at a receiver, if the receiver is not receiving any

other packets, it can receive this packet successfully. Thus, collision avoidance is entirely probabilistic [7].

One critical problem, however, which forms the focus of this study, is how to provide high performance MAC in UWSNs. In the literature, extensive studies have recently been conducted which have explored underwater MAC protocols. Because contentions-based MAC protocols are costly in UWSNs, collision-free MAC protocols promise to achieve better performance. However, the long propagation delay usually means that a centralised MAC protocol takes a long time to collect the global topology and transmission requests from all the sensor nodes and notify them of the schedule, thus a distributed solution is preferred.

In this paper, we propose a new protocol, called GC-MAC, which takes the throughput, energy efficiency, and fairness into consideration. GC-MAC performs graph colouring to achieve as many conflict-free transmissions as possible across the network. By scheduling the transmissions and receptions of data packets at both the sender and receiver sides, nodes can properly achieve the objectives of high throughput, energy efficiency, and fairness. GC-MAC uses the concept of graph colouring to develop a reservation-based contention-free MAC protocol. This is initially performed by utilising a distributed clustering approach for up to two-hop neighbouring nodes and then to address the near-far effect as well as the hidden and expose node problems by removing the possible colour conflict in two-hop neighbouring graph. Using a TDMA-like approach, GC-MAC is able to assign time slots to every individual colour in the network in a distributed manner. Nodes with the same colours can thus transmit concurrently without collision to support spatial reuse. Nodes are awake in some slots to transmit or receive data and asleep over the remaining slots.

The rest of this paper is organised as follows: In Section II, we review the related work. In Section III problem statements are illustrated. In Section IV, GC-MAC protocol is described in details. In Section V, the performance of GC-MAC protocol is evaluated and compared against those of ED-MAC [8], T-Lohi [9], and UWAN-MAC [10] protocols through simulations. Finally, Section VI concludes the paper.

## II. RELATED WORK

As the transmission of data packet consumes more energy in UWSNs, all existing MAC protocols attempt to avoid packets collisions and retransmissions. Despite these issues,

underwater MAC protocols can be divided into two categories: contention-free and contention-based.

The contention-based category can further be classified into two classes: random access-based and handshake-based MAC protocols [7, 11]. In the first class, random access-based, the protocols are modified ALOHA protocols. A short tone or preamble is used as a transmitting notification to neighbouring nodes. Upon receiving the transmitted notification of other nodes, it will back off its own transmission randomly or reschedule its own transmission based on the information from all its neighbouring notifications [12]. In this class, some random access-based MAC protocols are specifically designed for UWSNs to achieve high performance, such as UWAN-MAC [10] and ALOHA-AN [12]. However, this class with such notification schemes wastes channel bandwidth and energy because the hidden terminal problem is not detected. In the second class, some handshake-based MAC protocols have been proposed to provide valid solutions addressing the hidden terminal problem by using RTS/CTS, such as S-FAMA [13] and R-MAC [14]. This kind of protocols exploits virtual carrier sense to save energy and avoid collisions. They also use short control packets, which lower their chance of collision in comparison to regular data packets. However, the network throughput is usually low because of the high delay in the handshaking class. Due to the unique characteristics of acoustic modems, the contentions by using RTS/CTS control packets become costly. Hence, random access-based and handshake-based MAC classes are not as efficient as expected [15].

Since contentions are costly in UWSNs, a collision-free MAC protocol usually employs scheduling or utilises cluster architecture to prevent collisions. Some scheduling-based MAC protocols guarantee collision-free transmissions [8, 16–18]. Among them, ED-MAC [8] employs a duty cycle mechanism by assigning time slots to every individual node in the network in a distributed manner. ED-MAC generates a collision-free schedule by introducing the concept of sub-slots into every slot to prevent collisions between two neighbouring nodes of another node with lower depth, and by doing so, it significantly improves the network performance. In ED-MAC, however, the number of slots are doubled per round to prevent any possibility of concurrent data transmission from nodes located outside a one-hop neighbourhood and the node within the neighbourhood. Consequently, on the one hand it can provide a collision-free schedule, but on the other, it faces inefficient channel utilisations.

ST-MAC [16] is also collision free. It constructs a conflict graph based on the global topology information. It is also considered as a collision-free schedule with the conflict graph, and it improves the network performance. However, ST-MAC is a centralised scheduling algorithm. To create the conflict graph it requires the global networks topology data, which is expensive to collect in UWSNs due to low transmission rates and long propagation delays. Moreover, the priority traffic load assumption is extremely strong because it is usually unknown at the time of deployment. In addition, ST-MAC permanently allocates slots in a batch to the links with the highest traffic

TABLE I: Notations

Terms	Definition
$v$	Underwater sensor node
$rp$	Reference point
$m$	Beacon packet
$p_c$	Colouring packet
$p_{up}$	Updating packet
$p_{out}$	outer updating packet
$N_g$	Neighbouring graph
$Neig-info$	A neighbouring information list
$SNR$	Signal-to-noise-ratio
$Col$	Colour of a node
$CH$	A cluster head
$CM$	A cluster member
$d(v_i, rp_j)$	A distance between a node and a reference point
$CL$	Colouring list
$T_i$	Length of the initial phase
$T_s$	Length of the scheduling phase
$R_t$	Length of each round
$D_r$	Data generated rate
$S_i$	The side of internal cube
$S_x$	The side of external cube
$T$	The diameter of each face of the cube
$N_s$	Number of slots
$S_l$	Length of each slot
$\Delta_{DataProp}$	Propagation delay
$R_{time}$	Receiving time of data packet
$G_t$	Guard time
$T_r$	Transmission range
$U_s$	Speed of sound in water
$P_{tx}$	Packet transmission duration
$t_{data}$	Duration of data packet

to maximise network throughput. This technique, however, impairs system fairness and starves some nodes .

Compared with the aforementioned scheduling algorithms, GC-MAC is a distributed algorithm and can construct collision-free scheduling without requiring nodes' traffic loads in advance nor any global topology information. GC-MAC can also guarantee collision-free transmission of data packets through scheduling, and can be used for multi-hop UWSNs.

However, several MAC protocols utilise cluster architectures to eliminate collisions [19–21]. In [19], for instance, neighbouring nodes are grouped into clusters using TDMA for clustering and CDMA for inter-cluster communications. Inter-cluster communications occur through nodes belonging to multiple clusters, which are required to provide multi-user receiver systems. These nodes can therefore simultaneously receive packets with different CDMA codes from multiple clusters. Nevertheless, as mentioned in [18], although CDMA helps to avoid collisions, it may impair the system performance in low bandwidth characterised UWSNs due to packet transmission delays which are considerably extended by the spreading factor. In addition, multi-user receiver systems significantly increase system complexity and cost.

In contrast to these cluster-based MAC protocols, GC-MAC can efficiently schedule inner and outer sensor nodes across the network without requiring CDMA modulation or multi-user receiver systems, and it is still able to guarantee collision-free transmissions through the scheduling phase.

### III. PROBLEM STATEMENTS

When designing resource-sharing schemes in underwater acoustic networks, it is essential to consider the particular characteristics of the channel, such as low available bandwidth, long propagation delay, and frequency-dependent attenuation. Those constraints highly affect the MAC layer protocol design due to the challenges described as follows:

1) *Hidden terminal problem*: Since some of the nodes are outside the transmission range of one another, transmissions from one sensor node of an area may not be received by other sensor nodes within the same area. As shown in Fig. 1, node A can hear both B and C, while B and C cannot hear each other. When C attempts to transmit while B is transmitting data to A, it considers the channel free, thus it transmits data. A collision between the data packets from B and C hence occurs. This is a hidden terminal problem which results in data collisions, low throughput, and high energy consumption.

2) *Exposed terminal problem*: When a sensor node is prevented from transmitting packets to other sensor nodes because of a neighbouring transmitter. This problem can be seen in Fig. 1, where node A and node D are within the range of node B, but node E is not. Moreover, node B and node E are within the range of node D, but node A is not. If node B wants to send a packet to node A, and at the same time node D wants to send a packet to node E, both of these transmissions could occur simultaneously, as the destinations are out of each others range. Consequently, as node B starts to transmit its packet to node A, node D hears the transmission from node B and discontinues its transmission to node E, because node D supposes a collision will occur, if it continues its transmission.

3) *Spatial-temporal uncertainty problem*: To solve the collision problem in terrestrial wireless networks, it is necessary simply to restrict the interfering nodes from transmitting simultaneously, since the propagation delay is negligible in this case. In UWSN, however, it is essential to consider the location and transmission time of the node due to the long propagation delay of acoustic media. The spatial-temporal uncertainty problem can be defined as 'two-dimensional uncertainty', which is characterised as follows:

- The collision in the receiver is dependent on the propaga-

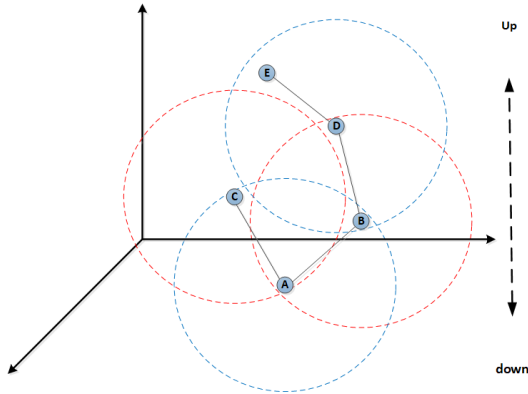


Fig. 1: An example of hidden terminal, exposed terminal, and spatial-temporal uncertainty problems

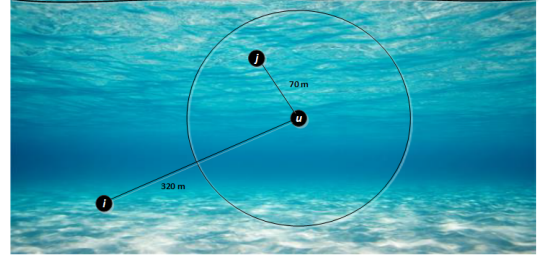


Fig. 2: An example of near and far problem

tion delay and transmission time and it can be shown as a duality that varies between both the transmission time and the location of the sensor nodes.

- The distance between the sensor nodes translates to uncertainty of current channel status and a packet may collide even if no other nodes send simultaneously.

Fig. 1, illustrates two cases of the spatial-temporal uncertainty problem. Firstly, when nodes B and C start transmitting to node A, their packets arrive at node A at different times because of different propagation delays. In the second case, when both nodes B and C transmit packets with different transmission times, collision might occur at node A.

4) *Near and far problem*: Due to the inherent characteristics of underwater acoustic channels as well as the CDMA technique, near-far effect is a major design challenge to the MAC protocols [22, 23]. If the received power for all sensors are not almost similar, signals from distant sensors cannot be received successfully. This is the near and far problem. This requires that the transmission power of each sensor must be controlled. Fig. 2 illustrates an example of both near and far effect. In this figure, the distance between  $i$  and  $u$  is almost four times or more than the distance between  $j$  and  $u$ . At the destination sensor  $u$ , therefore, signal-to-noise ratio (SNR) level of the signals originate from sensor  $j$  is higher than that from sensor  $i$ . This is mainly because of high level of noise produced by sensor  $j$ 's signals.

### IV. UNDERWATER GRAPH COLOURING (GC-MAC) PROTOCOL PRINCIPLE AND DESIGN

This section presents the system assumptions used in the GC-MAC protocol. We then exhibit the basic idea of our GC-MAC protocol. The clustering model used in the GC-MAC protocol is explained followed by a description of all the phases of GC-MAC protocol in detail.

#### A. System Assumptions

The following assumptions, which are widely used in the literature, are considered:

- All sensor nodes are randomly and uniformly distributed in a 3D underwater environment.
- A single sink is on the water surface, which is equipped with an acoustic modem for underwater communication as well as a radio modem for out-of-water communication.
- Anchored nodes are located at the bottom of the water to collect information which is then delivered to the sink

through relay nodes, which are located at different levels between the anchored nodes and the sink.

- Both anchored and relay nodes use acoustic signals to transmit packets.
- Each sensor node is assigned with a trio of coordinates  $(x, y, z)$ .
- All sensor nodes are homogeneous in terms of energy consumption and transmission range.
- Every sensor has a unique identifier (ID).

### B. Overview of GC-MAC protocol

To properly address the problems with utilising MAC protocols in UWSNs, such as hidden and expose node problems, near-far effect, and also the spatial-temporal uncertainty problem, we use the concept of graph colouring to develop a reservation-based contention-free MAC protocol. This is initially performed by utilising a distributed clustering approach for up to two-hop neighbouring nodes and then to resolve the near-far effect and also the hidden and expose node problems by removing the possible colour conflict in two-hop neighbouring graph. Using a TDMA-like approach, GC-MAC is able to assign time slots to every individual colour in the network in a distributed manner. Nodes with the same colours can thus transmit concurrently without collision to support spatial reuse. The primary goal is to reduce energy consumption by assigning each colour to a unique timeslot; nodes are awake in some slots to transmit or receive data and asleep over the remaining slots. GC-MAC trades off latency for high throughput, energy efficiency, and fairness, hence it is flexible enough to be utilised for various energy-critical applications in UWSNs.

Nodes in the network operate in three phases; namely initial, scheduling, and operational. All network nodes operate asynchronously during each phase, but share a common clock to start and end each phase together. To eliminate the effect of any clock drift that may occur over a long period of time, a guard time is applied.

The goal of the initial phase is to gather information of up to two-hop neighbouring nodes. This is performed by exchanging some beaconing packets. The length of this phase,  $T_i$ , is a predefined fixed value for all nodes, which is set at each node before deployment. It should also be noted that for highly mobile scenarios, the total length of initial, scheduling, and operational phases should be shorten to quickly react to topology changes in the network.

The purpose of the second phase, scheduling phase, is to assign different colours to all nodes which are located within any two-hop neighbourhood using a simple a clustering approach. Accordingly, a different colour is assigned by each cluster head to all one-hop neighbouring (inner) nodes. Then, the outer nodes decide about their own colours individually. By the end of this phase, every node has a different colour in any two-hop neighbouring graph and hence no collision can happen.

The operational phase is divided into several rounds; each round consisting of a number of slots. These slots are reserved

after assigning a unique colour to each slot. In other words, each colour represents a specific slot, as in conventional graph colouring, where the optimum spatial reuse can also be achieved by using the minimum number of colours. The length of each slot is equal to a signal propagation delay plus a small guard time. According to the schedule, all nodes should wake up either to transmit their own data packets during the reserved slots or to possibly receive data packets from neighbouring nodes. They are asleep in slots when there is no data transmission or reception.

### C. Network Model

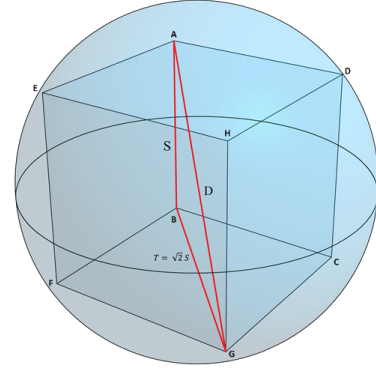


Fig. 3: Internal cube model of GC-MAC protocol

To facilitate distributed MAC scheduling, some reference points,  $rps$ , should be determined, the locations of which are known to all underwater nodes. Prior to the deployment process, therefore, the underwater area is divided into a number of adjacent cubes in order to set a reference point at the centre of each cube. Every cube is also encompasses a smaller co-centred cube. Nodes inside the internal cube are called inner nodes and nodes outside of the internal cube, but still inside the external cube, are called outer nodes. This is used to divide nodes into two groups depending on their distances from the associated reference point.

All vertices of the internal cube touch the surface of a sphere, as depicted in Fig. 3. The diameter of this sphere is the diagonal of the internal cube which is equal to  $2 \times T_r$ , where

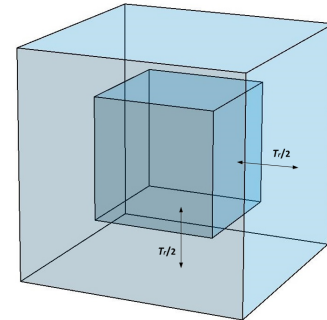


Fig. 4: The position of the internal cube in relation to the external cube.

---

**Algorithm 1: One-hop & two-hop beaconing**

---

```
1 Procedure Broadcast One-hop Info(node)
2 // type = 0 means (1-hop beacon type)
3 m: a new beacon message
4 if beacon timeout expired then
5   m.type ← 0
6   m.ID ← ID(node)
7   m.d ← d(node, rp)
8   Broadcast m
9   Set a new timeout
10 end procedure
11 Procedure Broadcast Two-hop Info(node, Ng)
12 // type = 1 means (2-hop beacon type)
13 m: a new beacon message
14 if beacon timeout expired then
15   m.type ← 1
16   m.ID ← ID(node)
17   m.Ng(ID, d) ← Neigh-info(node, ID, d)
18   Broadcast m
19 end procedure
```

---

$T_r$  is the transmission range of a sensor node. Each edge of the internal cube,  $S_i$ , can be calculated as

$$S_i = \frac{2 \times T_r}{\sqrt{3}} \quad (1)$$

Also, the diameter of each face of the cube,  $T$ , can be calculated by using the following equation

$$T = \sqrt{2} S_i \quad (2)$$

The distance between two sides of cubes is considered as ( $T_r/2$ ) in each direction, as shown in Fig. 4. Thus, the edge of the external cube,  $S_x$ , is given by

$$S_x = S_i + T_r \quad (3)$$

Finally, the distance between adjacent reference points is considered as  $S_i$  which is already calculated by equation 1.

In our model, the distance between a node  $v_i$  and its reference point  $rp_j$  is defined by using the coordinate of node  $v_i$  which is  $(x_i, y_i, z_i)$  and the coordinate of the reference point  $(rp_{jx}, rp_{jy}, rp_{jz})$ . Therefore, the distance between  $v_i$  and  $rp_j$  is given by

$$d(v_i, rp_j) = \sqrt{(rp_{jx} - x_i)^2 + (rp_{jy} - y_i)^2 + (rp_{jz} - z_i)^2} \quad (4)$$

using its own coordinates and the coordinates of the associated reference point, every sensor can determine whether it is an inner or outer node. This is required as different colouring algorithms are used for inner and outer nodes in the second phase.

#### D. Initial phase

At the deployment process, the start time of the initial phase for all sensor nodes has been set. During the initial phase, two rounds of beaconing are performed to obtain two-hop neighbouring information. In the first round, all the nodes randomly broadcast a few small beacons to discover their one-hop neighbouring nodes. Each node extracts the ID and distance to reference point from all packets received from neighbouring

---

**Algorithm 2: Inner colouring & updating**

---

```
1 Procedure Inner colouring(node)
2 A ← { a | a is the node with lowest distance to rp }
3 if (node == A) then
4   F(ID, Col) =  $\phi$  // start with empty set
5   CH ← node
6   pc: a new colouring packet
7   Colour(CH) ← First-Colour
8   F(ID, Col) ← F(ID, Col) ∪ Colour(CH)
9   for n ∈ neighbours(CH) do
10    Colour(n) ← Colour-Selection(n)
11    F(ID, Col) ← F(ID, Col) ∪ Colour(n)
12  pc.CL(ID, Col) ← F(ID, Col)
13  Broadcast pc
14 end procedure
15 Procedure Inner updating(node, pc)
16 if node ∈ pc.CL(ID, Col) then
17   Get Colour(node) from pc.CL(ID, Col)
18   pup: a new updating packet
19   F(ID, Col) = pc.CL(ID, Col)
20   for n ∉ neighbours(node) do
21    F(ID, Col) ← F(ID, Col) − n(ID, Col)
22   pup.CL(ID, Col) ← F(ID, Col)
23   Broadcast pup
24 end procedure
```

---

nodes and maintains them as one-hop neighbouring graph. In the second round, all the nodes randomly broadcast their one-hop graph to recognise their two-hop neighbouring nodes. The pseudocode for exchanging the two-hop neighbouring information is shown in Algorithm 1. The length of the initial phase,  $T_i$ , is a predefined fixed value for all nodes, which is set at each node before deployment. It should be long enough to allow the neighbouring nodes to create their own two-hop neighbouring graphs with accurate information. The length of this phase is very short compared to that of the third phase, however.

#### E. Scheduling phase

In this phase, two rounds are performed to assign colours to all inner neighbouring nodes and then to let the outer nodes selecting their own colours individually. The scheduling phase only requires to be repeated when the network topology changes.

1) *Inner nodes colouring*: At the beginning of this round, cluster heads (CHs) should be determined to be able to assign different colours to its inner nodes cluster members (CMs). A CH is simply the closest node to a reference point, which is recognisable by all nodes after the first phase. Every CH independently selects a colour for itself and provides different colours for all two-hop neighbourhood (inner) sensors by broadcasting a colouring packet. In this way, every inner node has a different colour in any two-hop neighbouring graph. The number of colours depends on the density of the sensor nodes in underwater area.

Upon receiving the colouring packet, all inner sensors obtain their specific chosen colours. The inner sensors then broadcast another packet called updating packet including its

---

**Algorithm 3: Outer colouring**

---

```
1 Procedure Outer colouring(node,  $p_{up}$ )
2 if node receives  $p_{up}$  and  $d(\text{node}, rp) > R$  then
3   Update  $N_g$  by  $p_{up}.CL(ID, Col)$ 
4    $p_{out}$ : a new outer updating packet
5   Add neighbours with colours from  $N_g$  to  $F(ID, Col)$ 
6    $p_{out}.CL(ID, Col) \leftarrow F(ID, Col)$ 
7   Broadcast  $p_{out}$ 
8   if  $N_g$  is fully updated by all neighbours colours then
9     node selects its colour based on the ID priority among
       neighbours
10 end procedure
```

---

ID, colour, neighbours' IDs, and neighbours' colours, to the outer neighbouring nodes. The purpose of sending this packet is to inform the outer nodes of the colours of their inner neighbouring nodes. Algorithm 2 describes how the inner nodes obtain their colours and how they inform their outer neighbouring nodes using updating packets.

2) *Outer nodes colouring*: Upon receiving the updating packet, every outer node removes non-neighbouring nodes from the list and then forward it to all other neighbouring nodes. Each outer node should then decide to select a colour with no conflict with any node located in two-hop neighbourhood, including the outer nodes belong to the adjacent cubes. This is accomplished by giving the higher priority to select the first available colour to the outer node with a lower ID. The ID of nodes has already been exchanged and every outer node is aware of its position among all two-hop neighbouring nodes. The pseudocode for exchanging the priority among the outer neighbouring nodes is shown in Algorithm 3. The length of this phase,  $T_s$ , is set to a predefined fixed value.

#### F. Operational phase

In this phase, nodes wake up and sleep periodically. This means that nodes are awake in some slots and asleep during the remaining slots when there is no data transmission or reception. This phase is divided into several rounds, each round consisting of a number of slots. These slots are reserved by assigning a different colour to each. In other words, each colour represents a different channel reservation as in conventional graph colouring, while optimum spatial reuse can also be achieved by using the minimum number of colours. Sensors with the same colour can concurrently transmit data packets without any collision while the near-far effect and the hidden and exposed terminal problems are properly addressed. The length of each slot is fixed and equal to a signal propagation delay plus a small guard time. The guard time is used to ensure that distinct transmissions do not interfere with each other. During each round, each node is aware of its own reserved slots based on its colour, as well as the slots reserved by its neighbouring nodes. They can therefore schedule to wake up either to transmit their own data packet during the reserved slots or to receive a data packet from a neighbouring node. They are asleep in other remaining slots when there is no data transmission or reception. This pattern is repeated during every

round. The length of this phase, i.e. the number of rounds, depends on topology changes due to node displacement or energy depletion. A shorter length should be considered for this phase in those scenarios with rapid topology changes and a longer period in scenarios with stationary or limited mobile nodes. Either way, it is a predefined fixed value configured on each node before deployment.

The length of each round,  $R_t$ , has a reverse relationship with data rate,  $D_r$ , which is presented in terms of packet per second. The higher the data rate, the shorter round time and, hence, the shorter the sleeping time. The duration of each round time is given by

$$R_t = \frac{1}{D_r} \quad (5)$$

Each  $R_t$  is divided into a number of equal size slots,  $N_s$ . The number of slots per round can be calculated by using the following equation

$$N_s = \frac{R_t}{S_l} \quad (6)$$

where  $S_l$  is the length of every slot, which is longer than the propagation delay to ensure that a packet is entirely received at the destination before starting of data transmission by another node. The length of each slot is given by

$$S_l = \Delta_{DataProp} + R_{time} + G_t \quad (7)$$

where  $R_{time}$  indicates the receiving time of data packet.  $G_t$  denotes the guard time, which is used to ensure that distinct transmissions do not interfere with one another. Also,  $\Delta_{DataProp}$  denotes the propagation delay of a transmitted packet, which can also be calculated using

$$\Delta_{DataProp} = \frac{T_r}{U_s} \quad (8)$$

where  $T_r$  denotes the transmission range and  $U_s$  indicates the speed of sound in water. Accordingly, the data rate,  $D_r$ , used in equation 5 should be limited to be used in our system model. To find its upper-bound, the following equation must be satisfied:

$$S_l \leq R_t \quad (9)$$

where the slot length,  $S_l$ , cannot be exceed the round time,  $R_t$ . The slot length is already calculated by equation 7. By replacing  $R_t$  using equation 5, it can be extended as

$$S_l \leq \frac{1}{D_r} \quad (10)$$

Based on the above equation, the upper-bound for  $D_r$  is calculated as

$$D_r \leq \frac{1}{\Delta_{DataProp} + R_{time} + G_t} \quad (11)$$

The upper-bound for  $D_r$  depends on the length of slot. The length of slot is a fixed value which is long enough to handle the consecutive receiving packets.

## V. PERFORMANCE EVALUATION

In this section, we first discuss the simulation specification of our GC-MAC protocol. We also evaluate important medium access metrics such as throughput, energy efficiency, and



fairness. We look at the design tradeoffs between GC-MAC and compared it with ED-MAC, T-Lohi, and UWAN-MAC protocols. We then present and analyse the simulation results.

### A. Simulation Specification

We implement GC-MAC in Aqua-Sim, an NS-2 based simulator for underwater sensor networks [24]. We perform simulations with the following parameters, unless otherwise noted. We randomly deployed nodes in a  $216\text{ m} \times 186624\text{ m}^2$  area for a fully connected network with acoustic transmission range of  $100\text{ m}$ . The bit rate for the acoustic modem is set to  $10\text{ kb/s}$  and packet length is  $1000\text{ bits}$ , implying that packet transmission duration  $P_{tx}$  is  $40\text{ ms}$ . The beaconing packet size is set to  $100\text{ bits}$ . The bandwidth is  $100\text{ Kb/s}$ . The power consumption on transmission mode is  $2\text{ Watts}$ ; the power consumption on receive mode is  $0.75\text{ Watts}$ ; and the power consumption on sleep mode is  $8\text{ mW}$ .

In our simulation, we evaluate three metrics of the medium access, which are throughput, energy efficiency, and fairness as a function of offered load. We also consider  $T_i$  as  $60\text{ seconds}$  and  $T_s$  as  $90\text{ seconds}$  in our simulation setup. In this set of simulations, we investigate the performance of GC-MAC protocol and compare it with ED-MAC, T-Lohi, and UWAN-MAC protocols.

### B. Performance Metrics

The performance metrics that we use are Throughput, Energy consumption, and Fairness index as a function of the offered load, defined as follows

$$\text{Offered load} = \frac{\text{Total generated packets} \times t_{data}}{\text{Simulation time}} \quad (12)$$

$$\text{Throughput} = \frac{\text{Total correct packets} \times t_{data}}{\text{Simulation time}} \quad (13)$$

where  $t_{data}$  denotes the duration of a data packet, which obtained by the corresponding propagation time plus the packet transmission time. The energy consumption is obtained by dividing the overall energy consumption in the network by the successfully delivered data packets, which is measured in joules per packet. Fairness index, however, is a key performance of MAC protocol, which affects the normal operation and survival time of UWSNs. To evaluate the fairness of among GC-MAC, ED-MAC, T-Lohi, and UWAN-MAC protocols, we adopt the Jain's Fairness Index define in [25]:

$$\text{Fairness index} = \frac{(\sum x_i)^2}{(n \cdot \sum x_i^2)} \quad (14)$$

where  $x_i$  denotes the throughput of node  $i$  and  $n$  denotes the number of nodes in the network. The ranges index value between 0 and 1 of a given metric could be used suitably as a measure of fairness, when the index value is closer to 1, this indicates that the protocol has a good fairness and vice versa.

### C. Simulation Results

The performance of GC-MAC is compared to that of ED-MAC, T-Lohi, and UWAN-MAC through simulations. For each test, the results are averaged over 50 runs, each obtained with a randomly generated topology in each run. The total simulation time for each run is set to 30 minutes. In this simulations, we evaluate throughput, energy consumption, and fairness of GC-MAC and compared it with three other MAC protocols as a function of offered load.

Fig. 5 shows that the achieved throughput of the proposed protocol is several times higher than T-Lohi and UWAN-MAC. The throughput of ED-MAC with low offered load reaches the same level of the proposed protocol, and then slightly raised as the load increases. Our proposed protocol outperforms ED-MAC, T-Lohi, and UWAN-MAC at the higher load by almost 38%, 79%, and 91% respectively. The throughput of both T-Lohi and UWAN-MAC is increased initially, but it degrades as the load increases. Both of them waste too much energy on collisions (over 50% unless the offered load is very low). This is because of more intensive channel competition and relatively increased the hidden terminal problems. Our proposed protocol is considered as a collision-free protocol. Taking into consideration that GC-MAC detects the near-far effect, the spatial-temporal uncertainty, and the hidden and exposed terminal problems where ED-MAC does not address the exposed terminal and near-far problems. Hence, when the load is increased, GC-MAC performs better than ED-MAC in terms of throughput.

In Fig. 6, the energy consumption of all MAC protocols is inversely proportional to the offered load. Our proposed protocol and ED-MAC are very efficient under all loads compared with other two protocols. This is mainly because of the hidden terminal problems which lead to more collisions and retransmissions. Our proposed protocol has a very low energy consumption because it prevents any data collisions. The ED-MAC energy cost increases marginally at higher loads nearly by 13% than that in GC-MAC due to the potential collision caused by two hidden nodes which are neighbouring nodes of another node with lower depth. It is more interesting to observe that GC-MAC and ED-MAC have nearly similar energy consumption during all offered loads. GC-MAC gets more packets through than ED-MAC, but ED-MAC has longer sleep periods during its operational window, thus the energy cost per packet becomes almost similar under the offered loads. The ED-MAC policy is to exclude the possibility of concurrent data transmission from nodes located outside of a one-hop neighbourhood and the node within the neighbourhood by doubled the number of slots per round, therefore it has longer sleep period than our proposed protocol.

Fig. 7 shows the fairness index of all four protocols. We first observe that GC-MAC protocol exhibits a high fairness index at almost 0.98 in lower load. Afterwards, it slightly decreases just over 0.90 within 4 offered load. As the offered load increases, the fairness index of all protocols declines considerably. This is because the network congestion reduces



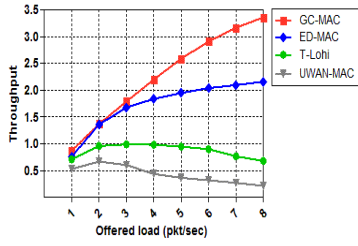


Fig. 5: Throughput vs offered load

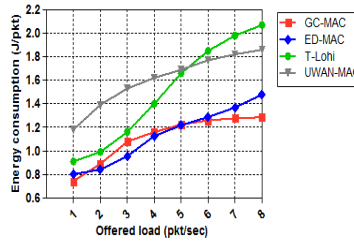


Fig. 6: Energy cost vs offered load

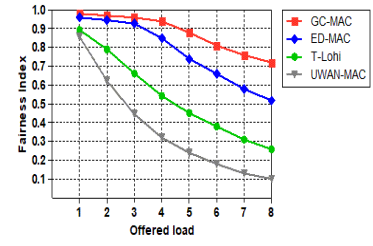


Fig. 7: Fairness index vs offered load

the fairness in all protocols. Due to the large delays in the underwater acoustic network, the distance between nodes becomes a key factor in the competitive channel. In comparison, GC-MAC and ED-MAC (within 2 and 3 offered loads) almost achieve the same fairness index at (0.95 and 0.94) respectively. When the loads further increases, GC-MAC reaches higher fairness than that in ED-MAC and other two protocols. Due to the considerably lower fairness of T-Lohi and UWAN-MAC compared to GC-MAC and ED-MAC, the temporal and spatial reuse are applied.

## VI. CONCLUSION

In this paper, we have proposed an efficient collision-free graph colouring MAC protocol (GC-MAC) for underwater sensor networks. GC-MAC uses a graph colouring approach with a duty cycle mechanism in order to assign each colour to a unique timeslot. Hence, nodes with the same colours can thus transmit concurrently without any collision. Using our proposed protocol, nodes are awake in some slots to transmit or receive data and asleep over the remaining slots. It is, therefore, able to reduce the energy consumption and to improve the throughput and fairness by handling the traffic contention effectively. The spatial-temporal uncertainty, near-far effect, and the hidden/exposed terminal problems have also been addressed. Using an extensive simulation study, the performance of GC-MAC has been compared against those of three other protocols from the same category recently reported in the literature. The results have shown the improvement achieved in terms of throughput, energy consumption, and fairness index with varying offered loads.

## REFERENCES

- [1] S. Xiong, C. Yuan, L. Tian, and Y. Zhan, "Ret-mac: A new fair mac protocol for underwater acoustic sensor network," *Jour. of Dist. Sen. Net.*, 2013.
- [2] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou, "The challenges of building mobile underwater wireless networks for aquatic applications," *IEEE Net.*, vol. 20, no. 3, pp. 12–18, 2006.
- [3] S. M. Ghoreyshi, A. Shahrabi, and T. Boutaleb, "An underwater routing protocol with void detection and bypassing capability," in *Advanced Info. Net.g and App. (AINA), 2017 IEEE 31st Int. Conf. on*, pp. 530–537, IEEE, 2017.
- [4] M. Stojanovic and J. Preisig, "Underwater acoustic communication channels: Propagation models and statistical characterization," *IEEE Comm. Mag.*, vol. 47, no. 1, pp. 84–89, 2009.
- [5] F. Alfouzan, A. Shahrabi, S. M. Ghoreyshi, and T. Boutaleb, "Performance comparison of sender-based and receiver-based scheduling mac protocols for underwater sensor networks," 2016.
- [6] D. Pompili, T. Melodia, and I. F. Akyildiz, "A cdma-based medium access control for underwater acoustic sensor networks," *IEEE Tran. on Wire. Comm.*, vol. 8, no. 4, pp. 1899–1909, 2009.
- [7] K. Chen, M. Ma, E. Cheng, F. Yuan, and W. Su, "A survey on mac protocols for underwater wireless sensor networks," *Comm. Surv. & Tutor., IEEE*, vol. 16, no. 3, pp. 1433–1447, 2014.
- [8] F. Alfouzan, A. Shahrabi, S. M. Ghoreyshi, and T. Boutaleb, "Efficient depth-based scheduling mac protocol for underwater sensor networks," in *Ubiqu. & Fut. Net. (ICUFN), Nin. Int. Conf.*, pp. 827–832, IEEE, 2017.
- [9] A. Syed, W. Ye, J. Heidemann, et al., "T-lohi: A new class of mac protocols for underwater acoustic sensor networks," in *INFOCOM 2008. The 27th Conf. on Comp. Comm. IEEE*, 2008.
- [10] M. K. Park and V. Rodoplu, "Uwan-mac: An energy-efficient mac protocol for underwater acoustic wireless sensor networks," *Ocea. Engin., IEEE Jour.*, vol. 32, no. 3, pp. 710–720, 2007.
- [11] I. F. Akyildiz, D. Pompili, and T. Melodia, "Challenges for efficient communication in underwater acoustic sensor networks," *ACM*, vol. 1, no. 2, pp. 3–8, 2004.
- [12] N. Chirdchoo, W.-S. Soh, and K. C. Chua, "Aloha-based mac protocols with collision avoidance for underwater acoustic networks," in *IEEE INFOCOM Int. Conf. on Comp. Comm.*, pp. 2271–2275, 2007.
- [13] M. Molins and M. Stojanovic, "Slotted fama: a mac protocol for underwater acoustic networks," in *OCEA. Asi. Pac.*, pp. 1–7, IEEE, 2007.
- [14] P. Xie and J.-H. Cui, "R-mac: An energy-efficient mac protocol for underwater sensor networks," in *Wire. Algor., Sys. and App., WASA 2007. Int. Conf.*, pp. 187–198, IEEE, 2007.
- [15] L. Hong, F. Hong, Z. Guo, and Z. Li, "Ecs: Efficient communication scheduling for underwater sensor networks," *Sensors*, vol. 11, no. 3, pp. 2920–2938, 2011.
- [16] C.-C. Hsu, K.-F. Lai, C.-F. Chou, and K. C.-J. Lin, "St-mac: Spatial-temporal mac scheduling for underwater sensor networks," in *INFOCOM, IEEE*, pp. 1827–1835, 2009.
- [17] J. Ma and W. Lou, "Interference-aware spatio-temporal link scheduling for long delay underwater sensor networks," in *Sen., Mesh and Ad Hoc Comm. & Net., Ann. IEEE Comm. Soc. Conf.*, pp. 431–439, IEEE, 2011.
- [18] K. Kreda II, P. Djukic, and P. Mohapatra, "Stump: Exploiting position diversity in the staggered tdma underwater mac protocol," in *INFOCOM 2009, IEEE*, pp. 2961–2965, IEEE, 2009.
- [19] F. Salva-Garau and M. Stojanovic, "Multi-cluster protocol for ad hoc mobile underwater acoustic networks," in *OCEANS 2003. Proc.*, vol. 1, pp. 91–98, IEEE, 2003.
- [20] K. B. Kreda II and P. Mohapatra, "A hybrid medium access control protocol for underwater wireless networks," in *Proceedings of the second workshop on Underwater networks*, pp. 33–40, ACM, 2007.
- [21] P. Wang, C. Li, and J. Zheng, "Distributed minimum-cost clustering protocol for underwater sensor networks (uwsns)," in *Comm., 2007. ICC'07. IEEE Int. Conf. on*, pp. 3510–3515, IEEE, 2007.
- [22] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad hoc networks*, vol. 3, no. 3, pp. 257–279, 2005.
- [23] J. Partan, J. Kurose, and B. N. Levine, "A survey of practical issues in underwater networks," *ACM SIGMOBILE Mobile Comp. and Comm. Review*, vol. 11, no. 4, pp. 23–33, 2007.
- [24] P. Xie, Z. Zhou, Z. Peng, H. Yan, T. Hu, J.-H. Cui, Z. Shi, Y. Fei, and S. Zhou, "Aqua-sim: an ns-2 based simulator for underwater sensor networks," in *OCEANS 2009, MTS/IEEE Biloxi-Marine Technology for Our Future: Global and Local Challenges*, pp. 1–7, IEEE, 2009.
- [25] R. Jain, D.-M. Chiu, and W. R. Hawe, *A quantitative measure of fairness and discrimination for resource allocation in shared computer system*, vol. 38. Eastern Research Laboratory, Digital Equipment Corporation Hudson, MA, 1984.